Deep learning as a tool for ecology and evolution

Marek L. Borowiec[1,2,*], Rebecca B. Dikow[3], Paul B. Frandsen[3,4], Alexander McKeeken[1], Gabriele Valentini[5], Alexander E. White[3,6]

1 Entomology, Plant Pathology and Nematology, University of Idaho, Moscow, ID, USA.
2 Institute for Bioinformatics and Evolutionary Studies (IBEST), University of Idaho, Moscow, ID, USA.
3 Data Science Lab, Office of the Chief Information Officer, Smithsonian Institution, Washington DC, USA
4 Department of Plant and Wildlife Sciences, Brigham Young University, Provo, UT, USA
5 School of Life Sciences, Arizona State University, Tempe AZ, USA
6 Department of Botany, National Museum of Natural History, Smithsonian Institution, Washington DC, USA

*** mborowiec@uidaho.edu**

Abstract

Deep learning is driving recent advances behind many everyday technologies, including those relying on speech and image recognition, natural language processing, and autonomous driving. It is also gaining popularity in biology, where it has been used for automated species identification, environmental monitoring, behavioral studies, DNA sequencing, and population genetics and phylogenetics, among other applications. Deep learning relies on artificial neural networks for predictive modeling and excels at recognizing complex patterns. Operating within the machine learning paradigm, deep learning can be viewed as an alternative to likelihood-based inference methods. It has desirable properties, including good performance and scaling with increasing complexity, while posing unique challenges such as sensitivity to bias in input data. In this review we provide a gentle introduction to deep learning, review its applications in ecology and evolution, and discuss its limitations and efforts to overcome them. We also provide a practical primer for biologists interested in including deep learning in their toolkit and identify its possible future applications.

Introduction

Ecology and evolutionary biology investigate complex patterns and processes. Because of this, a mathematical toolkit has been necessary to describe and explain fundamental components of organic evolution and ecological interactions such as inheritance, natural selection, adaptation, population dynamics, and food webs (Otto and Day, 2011). Today, mechanistic modeling of increasing complexity allows us to sequence and assemble genomes, identify traits under selection, model extinction and adaptation dynamics, and assess wildlife populations, to name just a few examples. Modern biologists are inundated with data, which aside from genetic sequences, also include digitized information on samples, specimens, and species. This wealth of data is driving the development of analytic tools that can provide new understanding, greater efficiency, and ease of use.

While the increasing amount of data allows unprecedented insight, it also makes practical aspects of ecological and evolutionary inference challenging. Complex patterns in large data sets are often better described with complex models and each new model needs to be developed by researchers. Moreover, likelihood-based mechanistic approaches designed to consider many variables can be so computationally expensive that they can no longer be applied to data routinely generated in modern studies. A promising alternative is likelihood-free inference, one example of which is machine learning. The goal of machine learning is to find a model that performs well at making predictions from the data. This contrasts with likelihood-based methods, which assume the model generating the data is known (Breiman, 2001). Broadly defined, machine learning has been in use for decades now for data transformations and clustering (e.g. Principal Component and Discriminant Function Analysis, K-means) and for optimization in much of model-based inference in ecology and evolution (e.g. Markov chains, genetic algorithms) (Sorensen and Gianola, 2002; Ghahramani, 2004; Mundry and Sommer, 2007; Reich et al., 2008; Hamblin, 2012). More recently machine learning has seen a dramatic surge in popularity with a slew of new algorithms and applications.

One of the approaches rapidly gaining popularity is deep learning. Deep learning relies on multilayered, connected processing units (artificial neural networks or ANNs) (Goodfellow et al., 2016). In the last ten years algorithmic developments, hardware improvements, and democratization of high-level software for building deep learning solutions contributed to a sudden rise in their popularity. Deep learning is at the core of emerging technologies such as self-driving cars and is responsible for significant improvements to widely-used information technology tools such as image and speech recognition or automated language translation (LeCun et al., 2015). These successes were possible because of a major advantage of deep learning over other machine learning approaches. Classical machine learning requires that important data features are first identified using expert domain knowledge (Guyon et al., 2008). This is a limitation where features adequately describing the data are not obvious or difficult to extract, as in images. Neural networks overcome this by discovering the most important data features and patterns relevant for the task at hand automatically. Researchers are now beginning to apply deep learning to problems across ecology and evolutionary biology, from community science projects and environmental monitoring through sequencing equipment output processing, to population genetics and phylogenetic inference. In this review we explain what neural networks are and how they work, summarize ecological and evolutionary biology problems they have been applied to, and give an overview of their promises and limitations. We also provide a primer for researchers who would like to apply deep learning to their own systems and questions along with references and resources for different platforms and particular applications.

What are neural networks and how do they learn?

There are several ways to describe what artificial neural networks are and how they are used as inference tools. Although the most obvious biological analogy is limited, it is helpful to visualize neural networks as computer algorithms inspired by the brain: composed of interconnected layers of nodes ("neurons") and connections ("synapses") capable of learning by changing how easy it is for neurons to fire and how strong the connections are (Box 2). Computers represent these layers and connections as matrices of numbers manipulated through linear algebra operations, ensuring that neural networks can be used with virtually any input that can be represented numerically (Figure 1). In mathematical sense, neural networks are simply a function mapping input onto a desired output. This general design is simple, but it makes neural networks extraordinarily powerful: a network with information flowing from input to output layer with at least one intermediate layer (i.e. feedforward network) can approximate any continuous function, regardless of its complexity (Cybenko, 1989; Hornik et al., 1989). These approximations can describe pixels of an image,

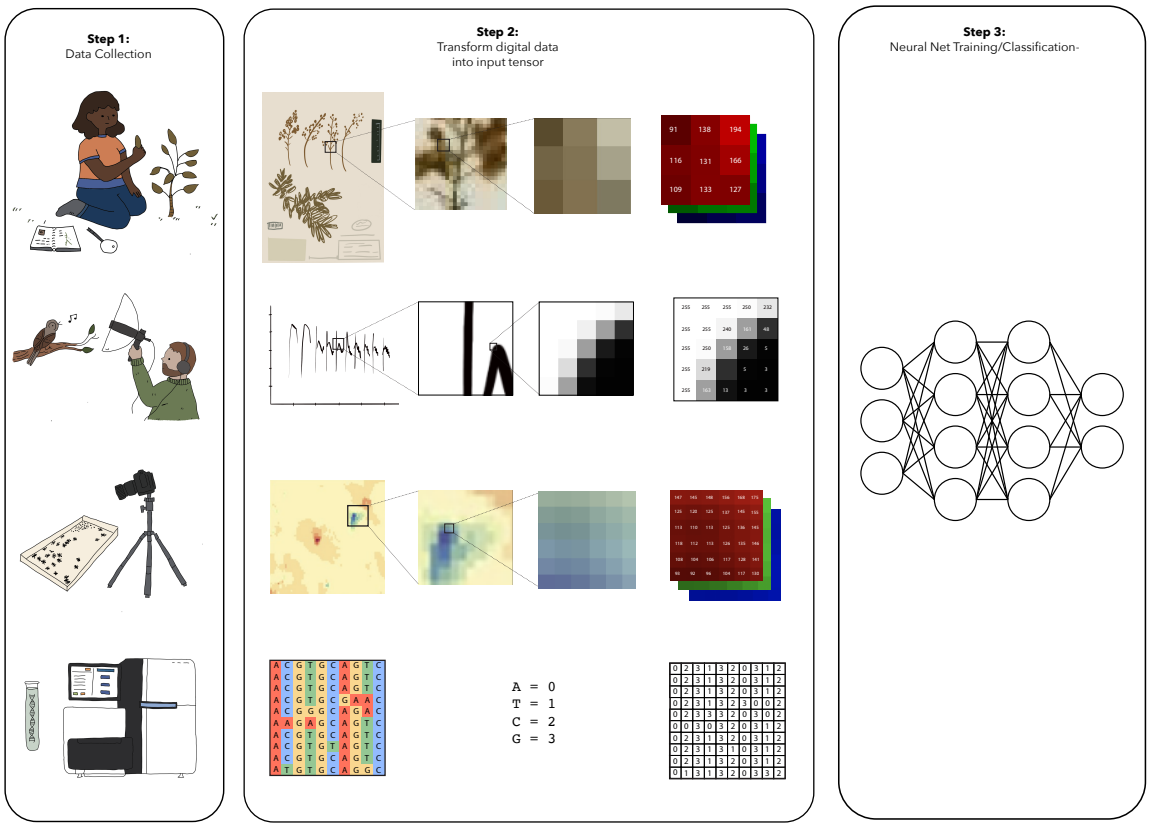


Figure 1. Workflow illustrating how various types of data can be encoded as input for deep learning. From top to bottom, the data sources include an image of a herbarium specimen, sonogram of a bird song, recorded insect motion represented as optical flow, and DNA data in the form of a multiple sequence alignment. Deep learning has been applied to all these data, which are here represented as either two- or three-dimensional input tensors. It is also possible to use one-dimensional (e.g. text string) or higher-dimensional input tensors (videos, 3D imaging). Tensor is a generalization of the term 'matrix', which is used for numbers arranged in a two-dimensional grid, to numbers arranged in any number of dimensions.

for example, and networks with multiple intermediate layers (deep neural networks) can also learn relationships between them as high-level concepts such as lines, geometric shapes, and even whole scenes (see Boxes 2 and 3). ANNs learn continuous distributions but their output can represent probabilities of distinct data classes, as well as continuous values. Such networks can thus be used to construct classifiers, which are models distinguishing among discrete categories, as well as regression models, which infer continuous values. However, feedforward operations alone do not allow the network to learn or generalize to new data, which is the essence of most ANN applications.

In order for an ANN to be a predictive tool, one needs to assess how good predictions are and be able to adjust ANN parameters to improve performance. A measure of how far off the output of the network is is called a loss function. One example of a loss function is the sum of squares error (SSE), which is simply the sum of differences between each predicted value (y) and the true value ($\bar{y}$) squared for absolute value:

$$SSE = \sum_{i=1}^{n} (y_i - \bar{y}_i)^2 \tag{1}$$

The network also needs a mechanism for finding the set of parameters that minimize the loss function. Once the loss (error) is measured at the output, it has to be traced back across the network to measure how parameters contributed to it. This process is called backpropagation and it uses chain rule calculus to find the derivative (slope) of the loss function with respect to the network's trainable parameters (LeCun et al., 1998, 2015). The process of increasing or decreasing parameters such that they minimize the derivative of the loss function is called gradient descent. Backpropagation and gradient descent illustrate the limits of the biological comparison, as no such learning mechanisms are known in biological nervous systems (but see Lillicrap et al. (2020)). This process is iterative, occurring every time a batch of training data is processed, and collectively referred to as the training loop (Figure 2). When devised correctly, it results in improvement of inference accuracy with each pass of the loop (Rumelhart et al., 1995).

The fact that ANNs are universal approximators for continuous functions that can be trained makes them powerful predictive tools. As we will see in the next section, ANNs are capable of solving problems that traditional computing has not been able to tackle. When compared to mechanistic, likelihood-based inference, the neural network training process is analogous to both model selection and parameter optimization. In contrast, neural network structure and choice of a particular loss function or optimizer algorithm are examples of hyperparameters. These are usually not learned from the data but are instead set using heuristics and optimized using trial-and-error for specific applications, although exceptions exist where hyperparameters themselves are optimized with deep learning (Cortes et al., 2017).

This learning scheme is most easily illustrated with an example of supervised learning, where the network is trained on a data set with ground truth known, e.g. images of expert-identified species or methylated vs. unmethylated DNA sequences. Supervised training usually involves splitting data into three subsets: training, validation, and testing. The validation set is not directly used in training but prediction on it is performed at the end of each training cycle (epoch) to assess how well the network generalizes outside of the training set. Because training is usually an iterative process, adjustments made between runs may lead to overfitting to the validation set. Therefore, test set is held back for the final estimate of accuracy.

Problems other than classification and regression in a supervised manner can be tackled with deep learning. Unsupervised learning is used where no ground truth is available, such as when visualizing patterns in data, clustering, or reducing dimensionality. The goal of network learning in this case is to reconstruct the original input after a transformation. The nature of this transformation will depend on the question asked. For example, a researcher may be interested in reducing genetic data to a few latent (hidden) variables that best describe geographic origins or population divergence of samples (Battey et al., 2020b,a). The training process will optimize network parameters such that the original input can be best reconstructed using these latent variables. Finally, reinforcement learning may be used to optimize actions of an agent interacting with a changing environment, rather than a fixed set of inputs. In this case, learning is determined by a goal or set of goals with feedback consisting of rewards and punishments. Reinforcement learning may be used to create more realistic models of ecological interactions and evolution, as was done in a predator-prey dynamics study where individual actors learn to hunt or avoid predators (Wang et al., 2019a).

In the sections that follow, we review how deep learning has been applied in ecology and evolution, including species identification and monitoring, ecological and behavioral studies, and population genetics and phylogenetics. We use these examples to showcase the variety of deep learning techniques extending its usage beyond the general picture described above. Figure 3 summarizes how different neural network

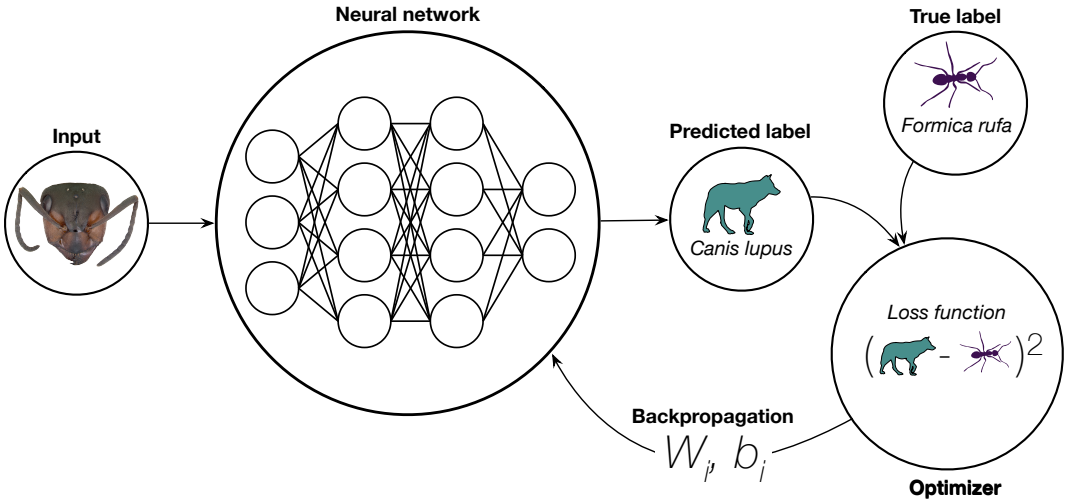


Figure 2. Deep learning classifier supervised training loop. Network weights and biases are random or initialized to certain values when the first batch of training data is input. The network predicts certain labels if it is a classifier, or continuous values if it is a regression model. This output is compared with true labels/values for that batch of input and a measure of error is computed using a loss function. The optimization procedure then works to adjust the network biases and weights in a way that improves output predictions. This is done with backpropagation or calculating derivatives of the loss function from the output layer back through the network and adjusting network parameters to minimize the error, a process known as gradient descent. This process is iterative, performed on each batch of training data. Input photograph of ant head by April Nobile, courtesy of www.antweb.org.

architectures have been applied to different tasks in ecology and evolution.

Automated species identification

Deep learning enabled breakthroughs in automated image classification, largely possible thanks to convolutional neural networks (CNNs; Box 3) (Goodfellow et al., 2016; Rawat and Wang, 2017). Image recognition has obvious applications in biology and was adopted early for problems of species identification and wildlife monitoring (see Wäldchen and Mäder (2018) and references therein). It is not surprising then that identification or classification of individuals or species from image, video, and sound data is the most common use of deep learning in the field (Figure 3). These efforts already span many taxa, from bacteria (Satoto et al., 2020), through protozoans (Hsiang et al., 2019), plants (Unger et al., 2016; Carranza-Rojas et al., 2017; Schuettpelz et al., 2017; Younis et al., 2018) to insects (Marques et al., 2018; Boer and Vos, 2018; Valan et al., 2019; Hansen et al., 2020) and vertebrates (Villon et al., 2018; Norouzzadeh et al., 2018), both extant and fossil (Liu and Song, 2020; Miele et al., 2020; de Lima et al., 2020) and at scales ranging from local to global. Intensifying efforts to digitize natural history collections provide troves of image data that can be used for this purpose (Smith and Blagoderov, 2012).

Camera trap systems and deep learning classifiers are now commonly used for vertebrate wildlife monitoring (Tabak et al., 2018) and systems automating environmental monitoring of aquatic macroinvertebrates are also being developed (Kiranyaz et al., 2011; Joutsijoki et al., 2014; Ärje et al., 2020; Høye et al., 2021). Many publications present systems or deep learning models for detecting and identifying pests or crop diseases in agroecosystems or stored agricultural commodities (Liu et al., 2016b,a; Ramcharan et al., 2017; Thenmozhi and Reddy, 2019; Wu et al., 2019). Despite economic importance and demonstrated potential for crop pest and disease monitoring, at present few non-proprietary systems or open source software applications exist (Kamilaris and Prenafeta-Boldú, 2018). A notable exception is a mobile application system for identifying diseases of cassava plants, one of the most important tropical crops (Ramcharan et al., 2019).

Deep learning has also been applied to identification from audio recordings, including bird and bat sounds, and even wing beats of mosquitoes (Fanioudakis et al., 2018; Mac Aodha et al., 2018; Stowell et al., 2019; Chen et al., 2020c). Unsurprisingly, the technology has been applied most often to bird calls,

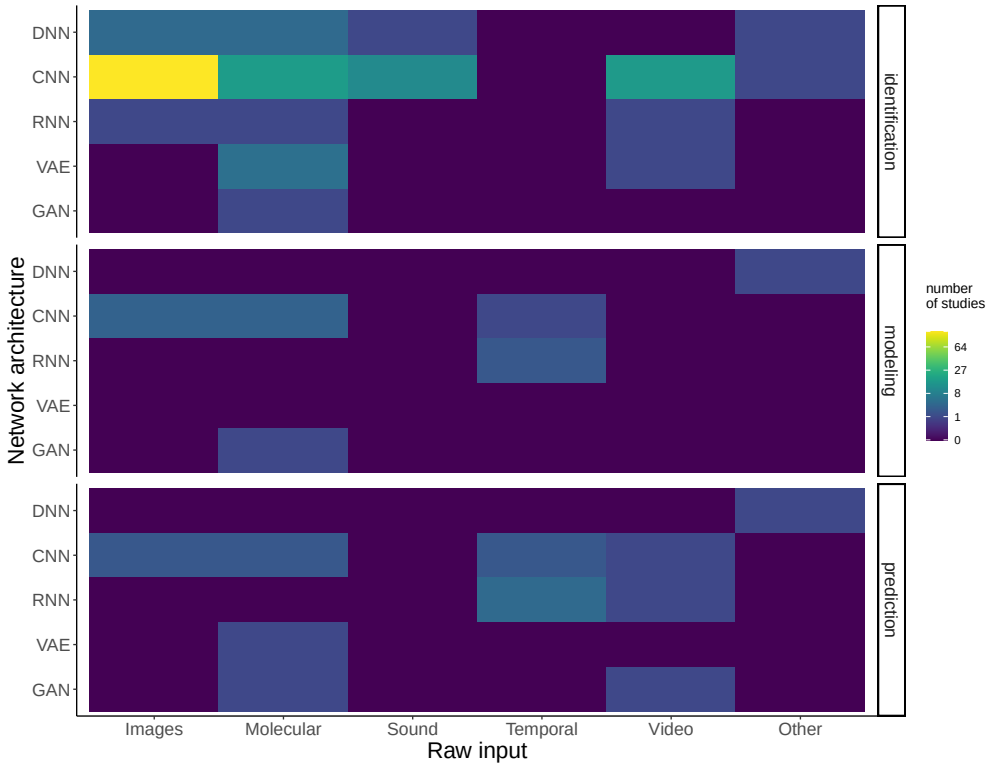


Figure 3. Deep learning approaches in ecology and evolution and their frequency of use by application. Panels represent three major applications: identification tasks, which include any studies classifying input into discrete categories; prediction tasks, or prediction of future events from time series and regression on continuous variables; and modeling, which includes simulating processes or phenomena. Rows in each panel correspond to different neural network architectures (see Box 2) and columns to input data prior to transformation. Based on 193 references in Supplementary Table, published through March 2021. Colors and legend numbers reflect number of studies. This figure shows that use of CNNs has been by far the most common (102 studies), and that they have been used primarily for image identification tasks but also using molecular data, sounds, and video. All architectures have been used with molecular data, highlighting versatility of deep learning in this area.

where it has been used not only to identify species, but also monitor their abundance (see Priyadarshani et al. (2018) for a review). The recently developed BirdNET is a deep neural network capable of identifying North American and European birds from vocalizations in complex soundscapes, available on a variety of platforms, including user-friendly smartphone apps (Kahl et al., 2021). Most of these studies use audio converted to spectrograms, image representations of sound, to train CNNs as in visual recognition problems.

Given its utility for automated identification, deep learning is increasingly used in community science initiatives. Examples include a growing number of mobile phone applications such as plant-focused identification app Pl@ntNet, bird identification tool Merlin or the citizen naturalist portal iNaturalist, as well as a number of more local or taxon-specific guides (Farnsworth et al., 2013; Wäldchen and Mäder, 2018; Sulc et al., 2020; Kahl et al., 2021). Many of these applications crowd-source training data collection and identification verification by users. They improve by periodically re-training their deep learning classifiers as more reliable data is collected.

Many of these studies employ data handling approaches that increase performance of deep learning classifiers. Several use data augmentation, a technique that relies on altering training data with distortion (Horn et al., 2018; Sulc et al., 2020; Kahl et al., 2021). These modifications, applied to each data input in each training epoch, effectively increase training set size. Data augmentation is an important strategy for reducing overfitting and almost always results in increased classifier accuracy (LeCun et al., 1998). By ensuring that the neural network never sees the same input twice, augmentation only partly addresses the fact that acquiring large, human-labeled datasets is a bottleneck for many applications. An alternative approach is to train an initial classifier in a supervised way, using a labeled training set, and then use this reasonably well-performing classifier for adding more images in an unsupervised manner, without human intervention (Rustia et al., 2019).

Another technique ubiquitous in identification and classification tasks is transfer learning (Chollet, 2018). Transfer learning is most commonly accomplished by first training on a different, usually larger and more general dataset than the one assembled for the problem on hand. The resulting network parameters can then be used as the starting point for fine-tuning on the focal dataset. In species recognition from images it is common to use networks pre-trained on large, public datasets of everyday objects such as ImageNet or COCO (Deng et al., 2009; Lin et al., 2014) as illustrated by several of the studies cited above (Norouzzadeh et al., 2018; Ramcharan et al., 2019; Sulc et al., 2020). Using pre-trained networks makes the network learn faster and often results in higher accuracy (Sharif Razavian et al., 2014).

In addition to properly assigning a label to an image, termed image classification, a common computer vision problem is to localize objects. Object recognition is a term often used for the combination of the two: drawing a bounding box around an object and predicting its class. Because there may be many objects in an image, this is a more challenging problem. The many proposed solutions involve either extracting candidate regions from images prior to prediction or predicting classes directly on grids of image pixels (Zhao et al., 2019). Examples are common in agriculture, where object detection has been used to identify and count pests (Ding and Taylor, 2016; Liu et al., 2016b; Fuentes et al., 2017; Shen et al., 2018; Zhong et al., 2018; Lins et al., 2020; Li et al., 2020a).

The deep learning framework allows training several neural networks of the same or varying architectures on one dataset and averaging their predictions. Known as model ensembling, this technique reduces variance in predictions and can improve accuracy (Goodfellow et al., 2016). Examples in species identification include Finnish fungi recognition and UK ladybird beetles (Sulc et al., 2020; Terry et al., 2020).

Finally, deep learning is not limited to considering image pixels alone but can also take advantage of contextual information such as locality or phenology. For example, output can be improved by filtering out nonsensical predictions given prior occurrence data (Berg et al., 2014; Wittich et al., 2018; Mac Aodha et al., 2019; Kahl et al., 2021). This approach, however, does not jointly consider the available data in a common framework. Neural networks can be trained on multiple data inputs simultaneously and consider them jointly in the final layers (Chollet, 2018). One study used this approach for beetle identification from images and found improvement in accuracy with information about location, date, weather, habitat, and user expertise (Terry et al., 2020).

Environmental monitoring and modeling

The above mentioned approaches to automated identification of species or individuals are also being scaled to ecosystem scale and applied to diversity assessment, conservation, and resource management (Christin et al., 2019). Examples using techniques detailed in the previous section include detecting and estimat-

ing abundance of zooplankton (Schmid et al., 2016) and detecting and counting sea turtles and whales using drone and satellite imagery (Gray et al., 2019; Guirado et al., 2019). Other uses combine digital imagery with LiDAR and other remote sensing or geospatial data for mapping of vegetation (Guo et al., 2020; Li et al., 2020b,c; Kislov and Korznikov, 2020; Korznikov et al., 2021), forest carbon stock (Asner et al., 2018), and the footprint of fishing across the world's oceans (Kroodsma et al., 2018). Similar applications include integrated systems for real-time wildlife monitoring using data from camera traps and microphones (Duhart et al., 2019) and raise the prospect of surveillance of social media posts for illegal animal trade (Minin et al., 2018).

In addition to classification and mapping of static information, recurrent neural networks (RNNs; Box 2) and similar approaches have been used with temporal ecological data. Examples include predicting eutrophication (Walter et al., 2001), phytoplankton blooms (Jeong et al., 2001, 2008; Ye and Cai, 2009; Malek et al., 2012; Srinivasan et al., 2018), and benthic invertebrate community dynamics (Chon et al., 2001). As mentioned previously, combining inputs from different sources is natural for deep learning and Rammer and Seidl (2019) take advantage of this to predict and map future bark beetle outbreaks based on temporal information on climate, vegetation, and past outbreaks. Capinha et al. (2020) proposed a generalized approach to classification and prediction from ecological time series data leveraging automated choice of the best network architecture for the task at hand.

Finally, neural networks are being used to develop more realistic models and simulations of real world patterns and phenomena. Benkendorf and Hawkins (2020) found that deep neural networks can be used to generate accurate species distribution models but also noted that other machine learning approaches perform as well or better with limited training data. Strydom et al. (2021) designed a system to predict species interactions from co-occurrence data. A study using reinforcement learning investigated how learning to hunt or avoid predators by individual agents influenced predator-prey dynamics (Wang et al., 2019a).

Behavioral studies

The study of animal behavior, both in the field and controlled laboratory settings, is another research area of ecology and evolution that is poised to greatly benefit from adoption of deep learning. Recent technological advancements in sensing, monitoring, and automation allow behavioral ecologists to collect and analyze large amounts of data (Egnor and Branson, 2016). Long-standing challenges in identifying, quantifying, and analyzing animal behavior still limit the ability to fully automate processing of these data, however. Deep learning has the potential to address many of these challenges and it is increasingly being adopted in studies involving identification of individual animals, body posture and movement tracking, and classification of behaviors.

In the area of animal body posture, deep learning can provide non-invasive estimation of the position of animals' body parts from video recordings (Mathis et al., 2020). Several open-source toolkits have been developed for this purpose, ranging from species-specific solutions (e.g., DeepFly3D for *Drosophila* (Günel et al., 2019), OpenMonkeyStudio for macaques (Bala et al., 2020)) to generic frameworks applicable to any species (e.g., LEAP (Pereira et al., 2019, 2020), DeepLabCut (Mathis et al., 2018; Nath et al., 2019), DeepPoseKit (Graving et al., 2019)), some of which offer 3-dimensional and/or multiple animals tracking. In addition to pose estimation, deep learning is also being adopted to enhance the performance of established computer vision methods used to track spatial position of animals (e.g., by tag detection (Sixt et al., 2018) or the identification of markers (Gal et al., 2020)), as well as to automatically perform behavioral analysis of spatial trajectories (Maekawa et al., 2020).

Deep learning can also allow for the identification, classification, and subsequent re-identification of individual animals from camera feeds or traps (Schneider et al., 2019), both in the wild and in captivity. Usually based on the use of CNNs for image recognition, deep learning can also be combined with other technologies (e.g. motion sensors, RFID) to develop automated data-processing pipelines to collect and label samples, as was done by Ferreira et al. (2020) for three different bird species. A popular application in this area is face recognition enabling mark-recapture studies for monitoring populations of individuals, their behavior, and social interactions. Examples in the wild include identification of elephants (Körschens et al., 2018), chimpanzees (Schofield et al., 2019), right whales (Bogucki et al., 2019), and brown bears (Clapham et al., 2020). Studies performed in captivity have been carried out on pandas (Chen et al., 2020b) and pigs (Hansen et al., 2018).

Finally, deep learning is being applied to automatically detect and classify the behavior of animals from raw data, a crucial step towards overcoming time-consuming and error-prone manual labeling tasks. Largely based on CNNs, a number of different so-

lutions have been developed to recognize and label behaviors from images (Norouzzadeh et al., 2018) as well as video (Fuentes et al., 2020; van Dam et al., 2020; Choi et al., 2021) and sound recordings (Bergler et al., 2019). These behavior detection systems can discriminate between behaviors (e.g. standing, resting, feeding, grooming), with the possibility of concurrent behaviors and thus multi-labeling, or be specifically designed to detect binary events (e.g. distinguish whale vocalizations from noise, Bergler et al. (2019), or rare social changes in otherwise stable insect colonies, Choi et al. (2021)). In addition to behavior recognition, deep learning solutions are also being devised to predict behavioral measurements that would otherwise require specialized recording devices. For example, Browning et al. (2018) used artificial neural networks to predict the diving behavior of seabirds from GPS data alone without specialized time-depth records, whereas Liu et al. (2019) used vertical movement sensors alone to predict locomotor energy expenditure of sharks.

Genomics, population genetics, and phylogenetics

A rapidly growing number of studies apply deep learning to study genomes (Zou et al., 2018). Deep learning is used in DNA sequencing for translating the raw signal of long-read Oxford Nanopore sequencers into nucleotide calls, outperforming other basecallers (Boža et al., 2017; Teng et al., 2018; Wick et al., 2019). Another example of successful application is variant calling, or identification of small nucleotide polymorphisms and indels in diploid or polyploid genomes. DeepVariant is a tool that converts text file representations of multiple sequences aligned to a reference (read pileups) to images and uses a CNN to predict alternative alleles (Poplin et al., 2018). Another tool predicts gene copy number variations from high-throughput sequencing reads (Hill and Unckless, 2019).

Deep learning has been particularly successful in functional and regulatory genomics and has been used for predicting sequence specificity of nucleic acid-binding proteins, methylation status, identification of transcription start sites, predicting expression patterns from genotypes, classification of transposable elements, and more. These applications are not strictly within the purview of ecology and evolution and have been comprehensively reviewed elsewhere (Park and Kellis, 2015; Angermueller et al., 2016; Zou et al., 2018).

Deep learning is a part of a growing trend to apply machine learning to the study of evolution of populations and species (Schrider and Kern, 2018). One of the early studies applying neural networks to population genetic data showed them capable of estimating population-scale mutation rates, population sizes and their changes through time, recombination rates, and detecting introgressed loci and positive selection on simulated data (Flagel et al., 2018). That study demonstrated that CNNs are capable of estimating population genetic parameters in scenarios for which likelihood-based methods have yet to be developed, such as accurately inferring recombination rates from read coverage data in autotetraploid genomes. The impressive performance of deep learning for population genetics encouraged recent development of user-friendly tools for inference from empirical data, including selective sweep classification (Kern and Schrider, 2018), quantifying selection strength (Torada et al., 2019), jointly inferring selection and population size change (Sheehan and Song, 2016), and inferring recombination landscapes (Adrion et al., 2020). Other studies relied on custom approaches to identifying deleterious variants in sorghum (Lozano et al., 2021) and positive selection in SARS-CoV-2 (Ouellette et al., 2021). An emerging approach involves combining deep learning with approximate Bayesian computation (ABC) (Beaumont et al., 2002; Bertorelle et al., 2010). It has been applied to inferring population size change through time (Sanchez et al., 2020), identifying hybridization from pairwise nucleotide divergences (Blischak et al., 2020), and choosing best-fitting demographic scenarios based on site frequency spectra or SNP data (Mondal et al., 2019; da Fonseca et al., 2020; Perez et al., 2020). Most of the above approaches use CNNs, which in their standard formulation are sensitive to permutations. This means that the ordering of chromosomes in the input, for example, is significant for training and prediction. Flagel et al. (2018) dealt with this by sorting chromosomes by similarity but network architectures insensitive to input ordering are also being developed (Chan et al., 2018).

Deep learning has also been used for inference and visualization of population structure (Derkarabetian et al., 2019; Battey et al., 2020b). Here neural networks are used for dimensionality reduction, similar to principal component analysis, rather than for solving a classification or regression problem. To achieve this, the authors used variational autoencoders (VAEs; Box 2), a pair of neural networks that learn efficient representations of data in an unsupervised manner. In this method the encoder network compresses input data into latent variables, while the decoder network attempts reconstructing the original data from those variables. The loss function in this case is a combined measure of how good the reconstruction is and de-

sirable properties of latent variables. In the case of (Battey et al., 2020b) and (Derkarabetian et al., 2019) the goal was to visualize population structure in a two-dimensional space and so the data were compressed into two variables representing coordinates.

As the importance of the spatial component is becoming increasingly highlighted in population genetics (Bradburd and Ralph, 2019), deep learning is also beginning to be used for predicting sample origins based on genetic variation (Battey et al., 2020a) and local-ancestry inference (Montserrat et al., 2020), which aims to identify populations from which a genetic locus descended. This application involves using generative adversarial networks (GANs; Box 2) (Wang et al., 2019b) to create artificial human genomic sequences of known ancestry (Montserrat et al., 2019; Yelmen et al., 2019).

In the field of phylogenetics, CNNs have been used for inference under conditions challenging existing likelihood-based methods. This includes correctly inferring topologies from alignments simulated under phylogenetic tree shapes known to produce biased results (Suvorov et al., 2019) and with substitution parameters changing across the phylogenetic tree (Zou et al., 2020). This work was motivated by the fact a majority of likelihood-based phylogenetic programs assume constant model parameters across the phylogeny, which can lead to biased inference (Foster, 2004). A recent application enables placement of new genetic sequences onto existing trees, a frequently needed but algorithmically challenging task (Jiang et al., 2021). Neural networks have also been used to identify whether the evolution of genetic markers collected from several species is better described by a bifurcating or reticulated phylogeny (Burbrink and Gehara, 2018) and to identify genetic locus properties contributing to gene-tree/species-tree discordance (Burbrink et al., 2019).

Limitations and how to overcome them

Important shortcomings of deep learning include sensitivity to quality and bias of training data and the "black box" aspect of machine learning. It is important to acknowledge these limitations and identify possible solutions.

Researchers using supervised deep learning are limited to datasets large enough for successful training. Training data may be unavailable, insufficient, or biased. It should also minimize noise and requires accurate labels. This is an obvious challenge for many systems and questions. Advances in digital technologies and automation continue to facilitate data acquisition and generation, however, as exemplified by the many studies cited in this review. Data quality can be improved by automation and standardization and, in some cases, augmentation with generative deep learning (Shorten and Khoshgoftaar, 2019). It is worth stating that there are many questions and datasets for which deep learning may not be the best tool. Complex patterns can be discerned with other machine learning approaches (Tarca et al., 2007; Schrider and Kern, 2018), although a comparative study seeking correlations between biological data properties and success of different machine learning algorithms is lacking.

A related problem involves using supervised deep learning for empirical problems for which no ground truth data exists. There are few population genetic or phylogenetic datasets, for example, for which we know the true generating evolutionary processes (but see Lenski et al. (1991); Hillis et al. (1992)). This does not prevent the use of supervised learning, however, because it is still possible to successfully infer parameters or scenarios in cases where simulation is feasible but where no likelihood-based methods for inference have been devised. This has been recently demonstrated for problems in population genetic inference (Flagel et al., 2018). Avoiding circularity in this approach requires understanding assumptions behind simulated training data and how it may depart from real life before drawing conclusions from empirical data. Asking how robust predictions are to model mis-specification, i.e. how unaccounted parameters may impact inference, should be standard practice for such studies.

Deep learning is also sensitive to systematic bias in training data (Kim et al., 2019). The well-documented over-representation of European males in human genomics datasets (Editorial, 2019), for example, impacts models designed to determine healthcare treatments (Kessler et al., 2016; Chen et al., 2020a). These biases not only impact the efficacy and robustness of the model to new data, but could also have impacts on clinical outcomes. For the types of data used in ecology and evolution, there are potential biases at every level of the data collection, model training, and evaluation process. For example, the way that natural history collections have grown over time is influenced by taxonomic and research interests, accessibility of collecting locations, perceptions of value to research programs or institutions, and political boundaries and the relationship of collections-based institutions to colonialism (Daru et al., 2018; Das and Lowe, 2018). Biodiversity data is therefore biased both taxonomically and geographically, which will im-

pact the inferences we can make about global patterns. The recent paper by Schell et al. (2020) demonstrated the ways ecology has been and continues to be shaped by systemic racism, which has ramifications for our natural history collections, species occurrence data, and environmental monitoring data. Such biases have the potential to impact traditional research methods, but because deep learning is often used to discern broad patterns across global datasets, any conclusions must be viewed through the lens of social and other factors that went into generation of the data. Training methods accounting for known data biases are actively pursued (Hendricks et al., 2018; Alvi et al., 2018), but they cannot address ignored or unknown biases.

Another potential source of bias can occur with the use of transfer learning. When pre-trained models are used, training data are often not openly available. We may not know the source(s) of the training data, how the data was collected, or how representative the different categories are. This has the potential to introduce bias into final models partly retrained with original training data, compromises ability to make the work truly reproducible, and has ethical implications for the downstream implementation. A solution to this is ensuring that deep learning workflows are adequately documented and adhere to the FAIR (findable, accessible, interoperable, and reusable) principles of data management from start to finish (Wilkinson et al., 2016).

In many of the applications described in this review, researchers generate biological insight from the final output of a neural network, with complex network architectures obscuring the path from data input to network output. For this reason, neural networks can be viewed as a "black box": a model effectively trained to parse biological data at fine resolution but the biological features guiding the model's decisions remain poorly understood. This perception is exacerbated by the "unreasonable effectiveness of deep learning" (Sejnowski, 2020): the method is simple to apply, yet broader questions about why deep learning is so successful are still unanswered (Sejnowski, 2020; Baraniuk et al., 2020). For practical applications in ecology and evolution (e.g., sorting data, identifying outliers, counting individuals or anatomical structures), an effective "black box" may be acceptable. However, scientific applications for hypothesis testing require a more complete interrogation of what information is learned and "represented" by a neural network and how that information is biologically relevant. The opaqueness may also contribute to hiding data biases highlighted above.

Indeed, deep learning is also referred to as "representation learning" (Bengio et al., 2013) because it generates a latent (hidden) space, in a fixed number of dimensions, in which the most relevant features of the data and the relationships between those features are represented mathematically. This latent space is generated (i.e. "learned") during network training, and the subsequent placement of the input data in the latent space is what is used to make predictions. Understanding the relationships between and among classes of data and the mechanisms that underlie those relationships is an important step for integrating deep learning into studies of ecology and evolution. Generating such insights requires simplifying complex neural network representations. Low dimensional projections of complex data are already afforded by more traditional tools in ecology and evolution (e.g. principal components analysis, multidimensional scaling, canonical correspondence analysis). These methods have long provided biological insights for data that were otherwise uninterpretable. Dimensionality reduction for a neural network architecture can be achieved by a number of methods (Van der Maaten and Hinton, 2008; McInnes et al., 2018; Szubert et al., 2019; Graving and Couzin, 2020). Other methods for interrogating neural network decisions may yield more fine-grained information. A recently developed "network dissection" shows how individual units within a neural network learn higher level conceptual information (Bau et al., 2020). Samek et al. (2021) describe 46 different methods for generating useful interpretations of neural networks (see Table 3 in Samek et al. (2021)). Such methods bridge the gap between highly accurate deep learning applications and the underlying features of the data that play a role in generating accurate predictions. These methods would find useful applications across various data types, allowing deep learning to drive insights in ecology and evolution.

Finally, it is worth noting there are no universal approaches to all inference problems (Wolpert and Macready, 1997) and deep learning is no exception. When considering a new approach, the researcher should establish a performance baseline against which to measure success and compare multiple models. Examples of problems not suitable for deep learning include questions for which analytic solutions exist, instances where mechanistic model itself is the focus, and datasets that are small and noisy.

Future directions

As studies applying deep learning to ecology and evolution continue to proliferate, we hope for increased scrutiny and mitigation of data biases and other common issues, some of which can be remedied by existing

best practices checklists (Roberts et al., 2021) and algorithmic auditing (Juneja and Mitra, 2021).

Combined with robotics, deep learning is likely to lead to increased automation in biodiversity discovery and monitoring and speeding up of digitization efforts. Future applications will move beyond distinguishing classes of data known from training to "open world learning", combining classification with outlier detection for the discovery of previously unknown data classes (e.g. phenotypes or genotypes) (Shu et al., 2018; Hassen and Chan, 2020). At the point of data collection, deep learning could eventually be used for on-the-spot detection of relevant samples (e.g. outlier genotypes, hybrids, cryptic species) to improve efficiency of field sampling efforts. The ability of neural networks to consider diverse data types could make studies combining molecular and phenotypic data easier in subfields that often consider them in isolation, such as in taxonomy and species delimitation. Ecology and evolution are also likely to see increased use of unsupervised learning for pattern visualization and discovery of gaps, as well as rise of new methods that interrogate neural networks for biologically meaningful features and their relationships.

In the fields relying on molecular data, deep learning has the potential to shortcut through the currently necessary multi-step linear analytic workflows. In phylogenetics, for example, this could be inference of trees directly from sequencing reads without the need for intermediate steps of data assembly and alignment (Guang et al., 2016).

Finally, we expect future integration of machine and deep training into the ecology and evolution graduate and postgraduate-level training, just as knowledge of statistics, chemistry, and bioinformatics have become standard components of most ecology and evolution programs.

A primer and computer system considerations

The deep learning technology is relatively accessible to computing-literate biologists, thanks to numerous learning resources and language platforms. Here we include a digital supplement with a computational notebook primer (Kluyver et al., 2016; Nelson and Hoover, 2020) describing three simple scenarios with biological data using Python libraries TensorFlow and Keras [link to Google Colab notebook supplement]. A primer on using deep learning to discover transcription-binding factors in DNA sequences accompanies the review by Zou et al. (2018). Beyond these tutorials, potential users should consider more comprehensive but still accessible hands-on primers by Chollet (2018) (for Python users) and Chollet and Allaire (2018) (for R users). A good way to develop intuition behind deep learning concepts is to consult the extensive visual introduction by Glassner (2021). Given the general interest in deep learning, a number of on-line resources is available, ranging from free tutorials to online graduate degree programs offered by major educational institutions.

Software frameworks for deep learning exist in most popular scripting languages, including Python and R, as well as MATLAB and Java. Python and R have the advantage of a large ecosystem of open-source tools for deep learning. The popular libraries include Caffe, Theano, Torch, and TensorFlow. Caffe is a C++ framework (Jia et al., 2014) which can be used as a standalone tool with command line interface or through a Python or MATLAB interface. Caffe was optimized for CNNs and image recognition problems. Theano (Bastien et al., 2012; The Theano Development Team et al., 2016) is a Python library that was designed as a flexible tool for defining mathematical expressions in a symbolic way and compiling them into highly efficient, hardware-specific implementations. Theano may be compelling for advanced users wanting to build neural networks declared as computational graphs. Torch was originally developed to be used with the scripting language Lua (Collobert et al., 2011) but has since been re-written with a popular and user-friendly Python interface as PyTorch (Paszke et al., 2019) and is also available for R (Falbel et al., 2021). A powerful feature of PyTorch is native support of structured data input, enabling easy creation of networks capable of processing inputs of variable shapes and sizes in the same training loop. Another strength is PyTorch's declarative data parallelism, allowing easy computing with multiple processors. TensorFlow (Abadi et al., 2016) is another library with a Python or R interface and currently the most popular deep learning framework. It offers a high-level API called Keras, which allows fast development with little code. Large user base and well-written series of official Keras tutorials (Chollet, 2018) make it easy to learn and find help on TensorFlow. TensorFlow supports static definition of neural network models, meaning that they need to be defined first and ran afterwards. This makes building networks capable of processing input of varying structure difficult compared to Torch. Data parallelism is also more challenging to implement although TensorFlow allows for great flexibility in this regard.

Deep learning relies heavily on linear algebra and computationally expensive matrix operations. Computer graphics is concerned with similar mathematics

and processors developed for optimum performance in that realm are called Graphical Processing Units, or GPUs. Until recently, commercial demand from the computer game industry was driving the development of powerful GPUs (Pratx and Xing, 2011) but the rising popularity of deep learning is disrupting this trend and now many GPUs are built with neural network training performance in mind. GPUs are one to two orders of magnitude more efficient than standard processors (CPUs) at deep learning computing. Individual researchers interested in applying deep learning to their research should therefore consider purchasing one, as even a mid-range gaming GPU will offer great advantage over almost any CPU. An alternative is to use on-line environments that make GPU resources available for limited time for free. An example is the Google Colab platform (Zou et al., 2018; Nelson and Hoover, 2020), which can be used for code prototyping. Users requiring long-term access to GPU resources may also consider purchasing cloud computing resources on one of the many well-documented platforms (Langmead and Nellore, 2018).

Conclusion

Deep learning can work on data from a variety of sources, learn complex, hierarchical patterns directly from raw input, and solve problems for which no likelihood-based solutions exist. These desirable qualities are motivating research that outweighs or overcomes weaknesses and challenges of deep learning. So far the most common application of deep learning has been to identify and classify in a supervised manner. In the future, classification tasks will further benefit from combining data from sources as disparate as images, motion and location sensors, and text annotations, to mention just a few possibilities. Increasing popularity and sophistication of sensor-based technologies in ecological research ensures that temporal prediction using deep learning is likely to proliferate. Unsupervised approaches to visualization and clustering of complex patterns such as population genetic structure are promising and will undoubtedly continue to be developed. Generative deep learning appears to be the best tool thus far for creating data with complexity approaching real-life biological patterns. As a competitor of and a complement to mechanistic modeling in the most complex inference problems, deep learning is on track to become an integral part of the future biologist's toolkit.

Box 1: Glossary

Architecture. The configuration of neurons, layers, and connections among them in a neural network.

Augmentation. A set of techniques for modifying existing training data in order to increase data set size for machine learning.

Backpropagation. A crucial method allowing **neural networks** to learn. Backpropagation measures the output error and adjusts network biases and weights in a way that minimizes it. This is done by computing errors starting with the output layer and deriving errors for neurons connected to it, then deriving errors for neurons for layers further back, all the way to the input layer, propagating the error backwards. Once an error is known, it can be minimized using **gradient descent**.

Batch. In the network **training** process, the number of input samples the network works through between computing error and updating weights and biases using **backpropagation**.
Class. A category of data in a discrete category classification problem.

Classifier. Algorithm mapping input variables to discrete categories, or target variables. A **neural network** distinguishing among different species captured on frames of a wildlife monitoring camera is a classifier. Contrast with **regression model**.

Convolutional Neural Network (CNN or ConvNet). A type of neural network commonly used in visual recognition, in which connections between different layers allow performing convolutions (see Box 3).

Dropout layer. Neural network layer that randomly masks a portion of outputs from another layer. Dropout is used to prevent **overfitting**.

Epoch. The number of complete passes of the **training set** in the network **training** process.

Filter. Also called a kernel, a filter is a **tensor** that is moved across the input to a **convolutional** layer, transforming it into a feature map. Through **training**, the values in the filter cells are optimized for detecting useful information, which in visual recognition tasks may include presence of horizontal or vertical lines or more complex patterns.

Gradient descent. Optimization algorithms used to find internal neural network parameters that improve performance according to a pre-defined measure of error. In other words, gradient descent is tasked with finding the best way to minimize the loss function. Many variations of gradient descent exist, designed to balance the goals of minimizing network learning time, avoiding local optima of parameter values, and improving overall accuracy.

Hidden layer. Layers of a **neural network** placed between input and output layers.

Loss function. A measure of error in **neural network** outputs.

Model. In the context of deep learning, the word "model" is often used to refer to a trained neural network. This can be confusing in studies that use neural network **classifiers** to distinguish among data generated under different evolutionary models or models of population demographics.

Object detection. Localizing, usually by surrounding with bounding boxes, and predicting the **class** of objects in images. Compare to **semantic segmentation**.

Overfitting. The error of a predictive model too narrowly fitting a limited set of data points. In the context of deep learning this means a network that performs well on the training and/or validation set but does not produce reliable results on other data.

Pooling layer. Neural network layer that reduces the feature map of a CNN (see Box 3). This is accomplished by sliding a window across the feature map **tensor** and calculating maximum or average value of cells in that window. The output serves to reduce feature map dimensions.

Glossary, continued

Pre-training. Training a **classifier** or **regression model** on data similar to **training set** before using any of the training data. Pre-training is used instead of initializing a model with random weights and biases, allowing for **transfer learning**.

Regression model. Algorithm mapping input variables to continuous target variables. A neural network estimating demographic parameter values of a population from genetic data is a regression model. Contrast with **classifier**.

Semantic segmentation. Predicting the **class** of objects for each pixel in images, giving information about object shape and the boundaries between overlapping objects. Compare to **object detection**.

Tensor. Mathematical term applied as a generalization of the term matrix, but with any number of dimensions. Example tensors include a zero-dimensional matrix or a single number, one-dimensional matrix or a series of numbers (often called a "vector"), two-dimensional matrix or numbers arranged into rows and columns, etc.

Test set. Data used to compare performance of a neural network beyond that used in **training** and **validation**. Although measures such as validation **loss** estimate how well the network generalizes, multiple rounds of training and adjustments to its parameters will cause **overfitting** to the validation set. Therefore, measures of accuracy should be performed on the test set only once.

Training. The process neural networks use to measure error and improve performance. Training happens in **batches**, where a pre-defined number of input samples is fed through the network, error is measured using the given **loss function**, and **backpropagation** and **gradient descent** are used to update the network such that is minimizes error. Complete training usually involves multiple iterations over the entire **training set**, or **epochs**.

Training set. Input data used directly for **training** of a neural network, as opposed to **validation** and **test sets**, which evaluate how well it generalizes beyond the training set.

Transfer learning. The process by which neural network can acquire skill from **training** on data similar, but not identical, to the problem at hand. This is achieved with **pre-training**.

Underfitting. The error of a predictive model failing to accurately capture the relationship between input data and target variables. In the context of deep learning this means a network that fails to produce reliable results on training as well as other data.

Validation set. Data used to evaluate the performance of a neural network on data beyond that used directly in **training**. The validation **loss** is computed at the end of each training **epoch** and is often the measure that decides training termination.

Box 2: Common neural network architectures

Neural networks (also known as Artificial Neural Networks, **ANNs**) are complex mathematical models composed of a large number of equations whose structure is represented as a network of interconnected nodes (neurons, one for each equation). Neurons are arranged in layers, can be connected to other neurons in different layers but not within the same layer. This arrangement of layers and connections is the network architecture. In the simplest form, a neural network has one input layer, at least one intermediate or hidden layer, and an output layer. In the so-called fully connected (or densely connected) layer, each node is connected with every node in the succeeding layer. The trainable network parameters consist of biases and weights. Each node has a bias value (b in panel A), which determine how easy it is for it to "fire". Each connection has a weight value (W) which represents connection strength. Each node also has an activation function (f), for example sigmoid function. Given some input (x), the so-called feedforward output (y) of a node is determined by a simple equation:

$$y = f(W \times x + b) \tag{2}$$

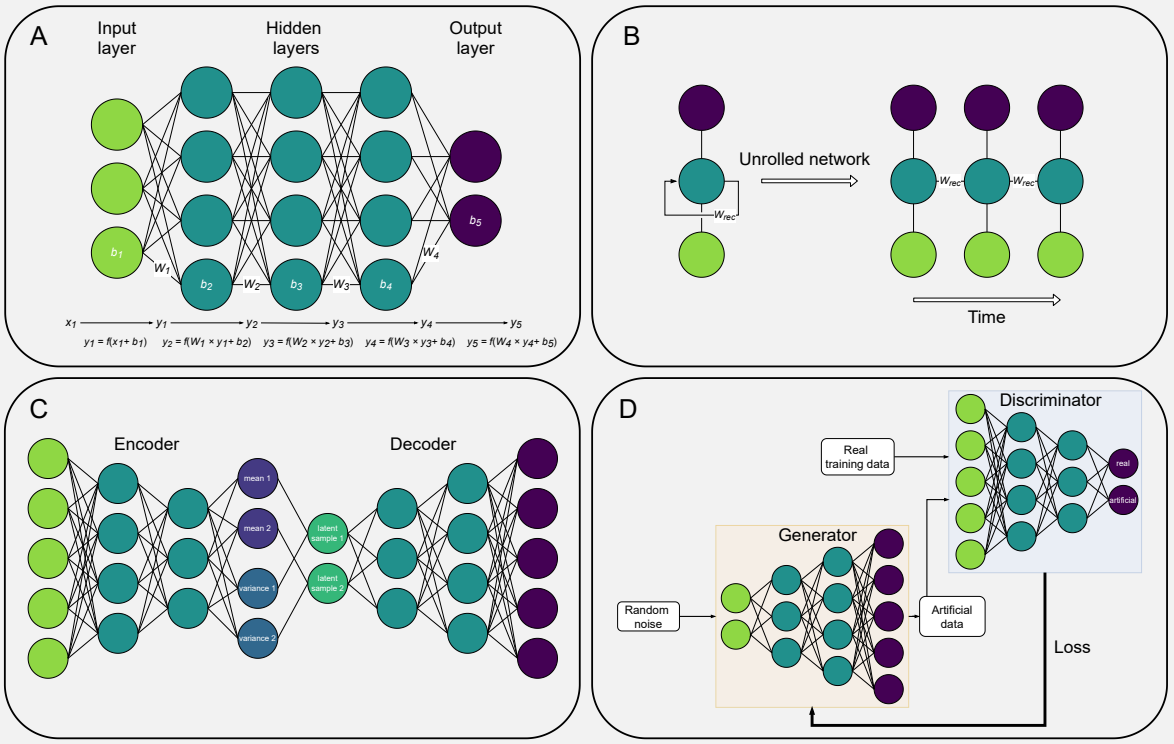
A fully connected or dense network (**DNN**; figure panel A), with information flowing from left to right. The number, arrangement, and type of layers and connections determine the neural network architecture. The original input (x_1) is progressively transformed (y_1 through y_5) as it moves forward through the network. In practice, values in neural networks are packaged in tensors (matrices of varying numbers of dimensions) and manipulated using linear algebra. For clarity, figure panel A shows only a single weight and input and output at one node per layer. In reality, outputs in this fully connected network would be calculated as a sum of all inputs and weights at each node. Although modern networks often use several different types of layers, fully connected layers are still used in many architectures. An important modification of the fully connected design involved rearranging the connections to allow for convolution operations described in Box 3. These convolutional neural networks (**CNNs**) excel at capturing complex, hierarchical patterns and are the architecture used in most identification and classification problems.

In dense networks, input data are usually shuffled prior to training and no information persists from one training batch to the next. This breaks relationships between consecutive data points and makes predicting events based on time series data impossible. Recurrent neural networks (**RNNs**; panel B) address this issue by adding loops to their information flow. A basic RNN can be imagined as a network with a single input, hidden, and output layer; Information flows from the input to the output of the network but it can also flow back from the output to the input of the hidden layer through recurrent weights (W_{rec} in panel B). A way to visualize this process is to "unroll" the network by connecting hidden layer neurons through time. The figure shows both representations. However, simple RNNs such as the one shown in the figure are difficult to train because weights in these networks can quickly diverge during training. More advanced derivations of the original RNN concept, such as Long Short-Term Memory networks (LSTMs) or Gated Recurrent Units (GRUs), address this problem and are commonly used with time series data or in language processing tasks. In evolutionary biology, deep learning solutions including GRU have been used to predict recombination landscapes.

A Variational Autoencoder (**VAE**; figure panel C) is another neural network architecture, composed of two parts: one called the encoder that maps input data onto a pre-defined number of latent variables, and one called the decoder that reconstructs the original input. Crucially, the encoder produces two vectors for each latent variable, one for the mean and the other for the standard deviation. This produces a continuous space for the latent variables. Thanks to this, VAEs are capable of generating new instances of data that are similar but not identical to the input. An example VAE use is visualization of population structure in two-dimensional space using genetic data as input. By transforming data into few, informative dimensions, VAEs are comparable to principal components analysis, but VAEs are additionally capable of non-linear dimensionality reduction.

A Generative Adversarial Network (**GAN**; figure panel D) is a type of neural network that generates artificial data by means of the interaction between two components: a generator and a discriminator. The generator produces plausible artificial data (i.e., similar to training set examples). The discriminator classifies the input as either real or artificial data. The generator takes random noise as its input and maps it onto the artificial output, which can be used along with real training examples as input for the discriminator. A variational autoencoder can be used as the generator. The decoder is trained to become better at distinguishing between real and artificially generated examples, whereas the loss function computed on artificially generated examples informs the training of the generator. This way the generator and discriminator improve each other's performance to produce data resembling real-world input. An example of GAN use is the generation of artificial genome sequences.

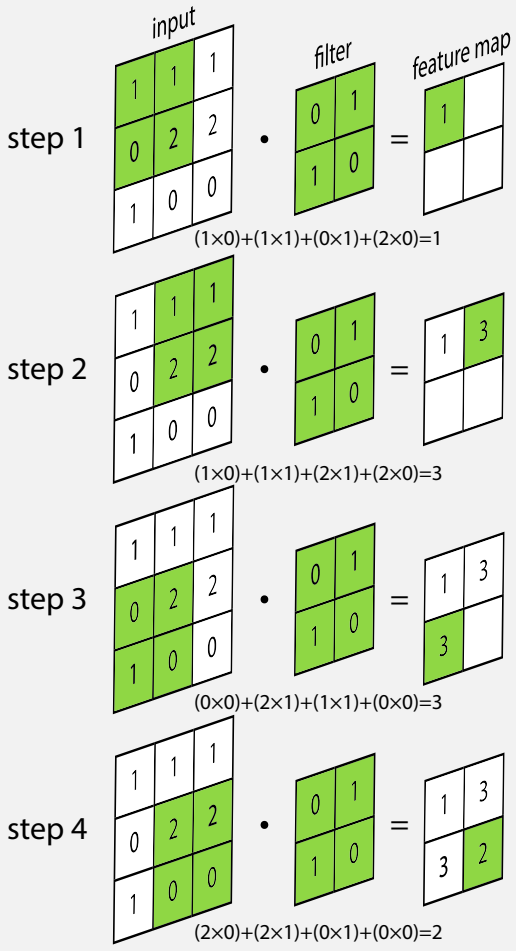
Box 2, continued



Box 3: Convolutions

A convolution is the operation of transforming an input tensor into an output called a feature map. It can be visualized as a window (called filter or kernel) sliding across the input in successive steps (see figure). Each step of a convolution takes the dot product between values in the input sector and those in the filter, resulting in a feature map. The filter consists of values that can be automatically learned by the network. In the figure, filter values correspond to network weights and the feature map calculations omit the bias term and activation function for clarity. Feature maps are crucial in visual recognition tasks because they capture information about the location of certain visual cues, for example, horizontal, angled, or vertical lines in an image, as well as more complex and abstract features. The process of detecting data features with a filter is often called "feature extraction". For example, the filter displayed in the figure is sensitive to diagonal lines and will produce feature maps with diagonal edges "extracted". Padding around the input may be used to preserve input dimensions in feature maps. There are usually multiple filters and corresponding feature maps in a single layer and networks commonly employ multiple stacked convolutional layers. In those cases, feature maps from the previous layer are used as input for the next layer, sometimes after passing through a pooling and/or dropout layer. Convolutional neural networks excel at capturing complex, hierarchical patterns.

Convolutions for image recognition are most often performed in two dimensions, as in the figure example, but they are also possible using data with different numbers of dimensions. For example, one-dimensional convolutions can be used on time series data or text strings and three-dimensional convolutions can be applied to video or 3D images.



References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). TensorFlow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283.

Adrion, J. R., Galloway, J. G., and Kern, A. D. (2020). Predicting the landscape of recombination using deep learning. *Molecular Biology and Evolution*, 37(6):1790–1808.

Alvi, M. S., Zisserman, A., and Nellåker, C. (2018). Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. *CoRR*, abs/1809.02169.

Angermueller, C., Pärnamaa, T., Parts, L., and Stegle, O. (2016). Deep learning for computational biology. *Molecular Systems Biology*, 12(7):878.

Ärje, J., Raitoharju, J., Iosifidis, A., Tirronen, V., Meissner, K., Gabbouj, M., Kiranyaz, S., and Kärkkäinen, S. (2020). Human experts vs. machines in taxa recognition. *Signal Processing: Image Communication*, 87:115917.

Asner, G. P., Brodrick, P. G., Philipson, C., Vaughn, N. R., Martin, R. E., Knapp, D. E., Heckler, J., Evans, L. J., Jucker, T., Goossens, B., Stark, D. J., Reynolds, G., Ong, R., Renneboog, N., Kugan, F., and Coomes, D. A. (2018). Mapped aboveground carbon stocks to advance forest conservation and recovery in malaysian borneo. *Biological Conservation*, 217:289–310.

Bala, P. C., Eisenreich, B. R., Yoo, S. B. M., Hayden, B. Y., Park, H. S., and Zimmermann, J. (2020). Automated markerless pose estimation in freely moving macaques with OpenMonkeyStudio. *Nature Communications*, 11(1):4560.

Baraniuk, R., Donoho, D., and Gavish, M. (2020). The science of deep learning. *Proceedings of the National Academy of Sciences*, 117(48):30029–30032.

Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D., and Bengio, Y. (2012). Theano: new features and speed improvements.

Battey, C., Ralph, P. L., and Kern, A. D. (2020a). Predicting geographic location from genetic variation with deep neural networks. *eLife*, 9:e54507.

Battey, C. J., Coffing, G. C., and Kern, A. D. (2020b). Visualizing population structure with variational autoencoders. *bioRxiv*, page e248278.

Bau, D., Zhu, J.-Y., Strobelt, H., Lapedriza, A., Zhou, B., and Torralba, A. (2020). Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078.

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035.

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828.

Benkendorf, D. J. and Hawkins, C. P. (2020). Effects of sample size and network depth on a deep learning approach to species distribution modeling. *Ecological Informatics*, 60:101137.

Berg, T., Liu, J., Lee, S. W., Alexander, M. L., Jacobs, D. W., and Belhumeur, P. N. (2014). Birdsnap: Large-scale fine-grained visual categorization of birds. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2019–2026. IEEE.

Bergler, C., Schröter, H., Cheng, R. X., Barth, V., Weber, M., Nöth, E., Hofer, H., and Maier, A. (2019). ORCA-SPOT: An Automatic Killer Whale Sound Detection Toolkit Using Deep Learning. *Scientific Reports*, 9(1):10997.

Bertorelle, G., Benazzo, A., and Mona, S. (2010). ABC as a flexible framework to estimate demography over space and time: some cons, many pros. *Molecular Ecology*, 19(13):2609–2625.

Blischak, P. D., Barker, M. S., and Gutenkunst, R. N. (2020). Chromosome-scale inference of hybrid speciation and admixture with convolutional neural networks. *bioRxiv*.

Boer, M. J. and Vos, R. A. (2018). Taxonomic classification of ants (Formicidae) from images using deep learning. *bioRxiv*.

Bogucki, R., Cygan, M., Khan, C. B., Klimek, M., Milczek, J. K., and Mucha, M. (2019). Applying deep learning to right whale photo identification. *Conservation Biology*, 33(3):676–684.

Boža, V., Brejová, B., and Vinař, T. (2017). DeepNano: Deep recurrent neural networks for base calling in MinION nanopore reads. *PLOS ONE*, 12(6):e0178751.

Bradburd, G. S. and Ralph, P. L. (2019). Spatial population genetics: It's about time. *Annual Review of Ecology, Evolution, and Systematics*, 50(1):427–449.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3):199–231.

Browning, E., Bolton, M., Owen, E., Shoji, A., Guilford, T., and Freeman, R. (2018). Predicting animal behaviour using deep learning: GPS data alone accurately predict diving in seabirds. *Methods in Ecology and Evolution*, 9(3):681–692.

Burbrink, F. T. and Gehara, M. (2018). The biogeography of deep time phylogenetic reticulation. *Systematic Biology*, 67(5):743–755.

Burbrink, F. T., Grazziotin, F. G., Pyron, R. A., Cundall, D., Donnellan, S., Irish, F., Keogh, J. S., Kraus, F., Murphy, R. W., Noonan, B., Raxworthy, C. J., Ruane, S., Lemmon, A. R., Lemmon, E. M., and Zaher, H. (2019). Interrogating genomic-scale data for Squamata (lizards, snakes, and amphisbaenians) shows no support for key traditional morphological relationships. *Systematic Biology*, 69(3):502–520.

Capinha, C., Ceia-Hasse, A., Kramer, A. M., and Meijer, C. (2020). Deep learning for supervised classification of temporal data in ecology. *bioRxiv*.

Carranza-Rojas, J., Goeau, H., Bonnet, P., Mata-Montero, E., and Joly, A. (2017). Going deeper in the automated identification of herbarium specimens. *BMC Evolutionary Biology*, 17(1).

Chan, J., Perrone, V., Spence, J. P., Jenkins, P. A., Mathieson, S., and Song, Y. S. (2018). A likelihood-free inference framework for population genetic data using exchangeable neural networks.

Chen, I. Y., Pierson, E., Rose, S., Joshi, S., Ferryman, K., and Ghassemi, M. (2020a). Ethical machine learning in health care.

Chen, P., Swarup, P., Matkowski, W. M., Kong, A. W. K., Han, S., Zhang, Z., and Rong, H. (2020b). A study on giant panda recognition based on images of a large proportion of captive pandas. *Ecology and Evolution*, 10(7):3561–3573.

Chen, X., Zhao, J., hua Chen, Y., Zhou, W., and Hughes, A. C. (2020c). Automatic standardized processing and identification of tropical bat calls using deep learning approaches. *Biological Conservation*, 241:108269.

Choi, T., Pyenson, B., Liebig, J., and Pavlic, T. P. (2021). Identification of abnormal states in videos of ants undergoing social phase change. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Chollet, F. (2018). *Deep Learning with Python*. Manning Publishing, Shelter Island, NY.

Chollet, F. and Allaire, J. J. (2018). *Deep Learning with R*. Manning Publishing, Shelter Island, NY.

Chon, T.-S., Kwak, I.-S., Park, Y.-S., Kim, T.-H., and Kim, Y. (2001). Patterning and short-term predictions of benthic macroinvertebrate community dynamics by using a recurrent artificial neural network. *Ecological Modelling*, 146(1-3):181–193.

Christin, S., Hervet, , and Lecomte, N. (2019). Applications for deep learning in ecology. *Methods in Ecology and Evolution*, 10(10):1632–1644.

Clapham, M., Miller, E., Nguyen, M., and Darimont, C. T. (2020). Automated facial recognition for wildlife that lack unique markings: A deep learning approach for brown bears. *Ecology and Evolution*, 10(23):12883–12892.

Collobert, R., Kavukcuoglu, K., and Farabet, C. (2011). Torch7: A Matlab-like environment for machine learning. In *BigLearn, NIPS workshop*, number CONF.

Cortes, C., Gonzalvo, X., Kuznetsov, V., Mohri, M., and Yang, S. (2017). AdaNet: Adaptive structural learning of artificial neural networks.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Control Signal Systems*, 2(4):303–314.

da Fonseca, E. M., Colli, G. R., Werneck, F. P., and Carstens, B. C. (2020). Phylogeographic model selection using convolutional neural networks. *bioRxiv*.

Daru, B. H., Park, D. S., Primack, R. B., Willis, C. G., Barrington, D. S., Whitfeld, T. J. S., Seidler, T. G., Sweeney, P. W., Foster, D. R., Ellison, A. M., and Davis, C. C. (2018). Widespread sampling biases in herbaria revealed from large-scale digitization. *New Phytologist*, 217(2):939–955.

Das, S. and Lowe, M. (2018). Nature Read in Black and White: decolonial approaches to interpreting natural history. *Journal of Natural Science Collections*, 6:4–14.

de Lima, R. P., Welch, K. F., Barrick, J. E., Marfurt, K. J., Burkhalter, R., Cassel, M., and Soreghan, G. S. (2020). Convolutional neural networks as an aid to biostratigraphy and micropaleontology: A test on Late Paleozoic microfossils. *Palaios*, 35(9):391–402.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255.

Derkarabetian, S., Castillo, S., Koo, P. K., Ovchinnikov, S., and Hedin, M. (2019). A demonstration of unsupervised machine learning in species delimitation. *Molecular Phylogenetics and Evolution*, 139:106562.

Ding, W. and Taylor, G. (2016). Automatic moth detection from trap images for pest management. *Computers and Electronics in Agriculture*, 123:17–28.

Duhart, C., Dublon, G., Mayton, B., Davenport, G., and Paradiso, J. A. (2019). Deep learning for wildlife conservation and restoration efforts. In *36th International Conference on Machine Learning, Long Beach*, volume 5.

Editorial (2019). Whose genomics? [Editorial]. *Nature Human Behaviour*, 3(5):409–410.

Egnor, S. R. and Branson, K. (2016). Computational analysis of behavior. *Annual Review of Neuroscience*, 39(1):217–236. PMID: 27090952.

Falbel, D., Luraschi, J., Selivanov, D., Damiani, A., and RStudio (2021). *torch: Tensors and Neural Networks with 'GPU' Acceleration*.

Fanioudakis, E., Geismar, M., and Potamitis, I. (2018). Mosquito wingbeat analysis and classification using deep learning. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 2410–2414. IEEE.

Farnsworth, E. J., Chu, M., Kress, W. J., Neill, A. K., Best, J. H., Pickering, J., Stevenson, R. D., Courtney, G. W., VanDyk, J. K., and Ellison, A. M. (2013). Next-generation field guides. *BioScience*, 63(11):891–899.

Ferreira, A. C., Silva, L. R., Renna, F., Brandl, H. B., Renoult, J. P., Farine, D. R., Covas, R., and Doutrelant, C. (2020). Deep learning-based methods for individual recognition in small birds. *Methods in Ecology and Evolution*, 11(9):1072–1085.

Flagel, L., Brandvain, Y., and Schrider, D. R. (2018). The unreasonable effectiveness of convolutional neural networks in population genetic inference. *Molecular Biology and Evolution*, 36(2):220–238.

Foster, P. (2004). Modeling compositional heterogeneity. *Systematic Biology*, 53(3):485–495.

Fuentes, A., Yoon, S., Kim, S., and Park, D. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 17(9):2022.

Fuentes, A., Yoon, S., Park, J., and Park, D. S. (2020). Deep learning-based hierarchical cattle behavior recognition with spatio-temporal information. *Computers and Electronics in Agriculture*, 177:105627.

Gal, A., Saragosti, J., and Kronauer, D. J. (2020). anTraX, a software package for high-throughput video tracking of color-tagged insects. *eLife*, 9:e58145.

Ghahramani, Z. (2004). Unsupervised learning. In *Advanced Lectures on Machine Learning*, pages 72–112. Springer Berlin Heidelberg.

Glassner, A. (2021). *Deep learning: a visual approach*. No Starch Press, San Francisco.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Graving, J. M., Chae, D., Naik, H., Li, L., Koger, B., Costelloe, B. R., and Couzin, I. D. (2019). Deepposekit, a software toolkit for fast and robust animal pose estimation using deep learning. *eLife*, 8:e47994.

Graving, J. M. and Couzin, I. D. (2020). VAE-SNE: a deep generative model for simultaneous dimensionality reduction and clustering.

Gray, P. C., Fleishman, A. B., Klein, D. J., McKown, M. W., Bézy, V. S., Lohmann, K. J., and Johnston, D. W. (2019). A convolutional neural network for detecting sea turtles in drone imagery. *Methods in Ecology and Evolution*, 10(3):345–355.

Guang, A., Zapata, F., Howison, M., Lawrence, C. E., and Dunn, C. W. (2016). An integrated perspective on phylogenetic workflows. *Trends in Ecology & Evolution*, 31(2):116–126.

Guirado, E., Tabik, S., Rivas, M. L., Alcaraz-Segura, D., and Herrera, F. (2019). Whale counting in satellite and aerial images with deep learning. *Scientific Reports*, 9(1).

Günel, S., Rhodin, H., Morales, D., Campagnolo, J. a., Ramdya, P., and Fua, P. (2019). DeepFly3D, a deep learning-based approach for 3d limb and appendage tracking in tethered, adult *Drosophila*. *eLife*, 8:e48571.

Guo, Q., Jin, S., Li, M., Yang, Q., Xu, K., Ju, Y., Zhang, J., Xuan, J., Liu, J., Su, Y., Xu, Q., and Liu, Y. (2020). Application of deep learning in ecological resource research: Theories, methods, and challenges. *Science China Earth Sciences*, 63(10):1457–1474.

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. (2008). *Feature Extraction: Foundations and Applications*. Studies in Fuzziness and Soft Computing. Springer Berlin Heidelberg.

Hamblin, S. (2012). On the practical usage of genetic algorithms in ecology and evolution. *Methods in Ecology and Evolution*, 4(2):184–194.

Hansen, M. F., Smith, M. L., Smith, L. N., Salter, M. G., Baxter, E. M., Farish, M., and Grieve, B. (2018). Towards on-farm pig face recognition using convolutional neural networks. *Computers in Industry*, 98:145–152.

Hansen, O. L. P., Svenning, J.-C., Olsen, K., Dupont, S., Garner, B. H., Iosifidis, A., Price, B. W., and Høye, T. T. (2020). Species-level image classification with convolutional neural network enables insect identification from habitus images. *Ecology and Evolution*, 10(2):737–747.

Hassen, M. and Chan, P. K. (2020). Learning a neural-network-based representation for open set recognition. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 154–162. SIAM.

Hendricks, L. A., Burns, K., Saenko, K., Darrell, T., and Rohrbach, A. (2018). Women Also Snowboard: Overcoming Bias in Captioning Models. In Ferrari, V., Hebert, M., Sminchisescu, C., and Weiss, Y., editors, *Computer Vision – ECCV 2018*, volume 11207, pages 793–811. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

Hill, T. and Unckless, R. L. (2019). A deep learning approach for detecting copy number variation in next-generation sequencing data. *G3: Genes, Genomes, Genetics*, 9(11):3575–3582.

Hillis, D., Bull, J., White, M., Badgett, M., and Molineux, I. (1992). Experimental phylogenetics: generation of a known phylogeny. *Science*, 255(5044):589–592.

Horn, G. V., Aodha, O. M., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., and Belongie, S. (2018). The iNaturalist species classification and detection dataset.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366.

Høye, T. T., Ärje, J., Bjerge, K., Hansen, O. L. P., Iosifidis, A., Leese, F., Mann, H. M. R., Meissner, K., Melvad, C., and Raitoharju, J. (2021). Deep learning and computer vision will transform entomology. *Proceedings of the National Academy of Sciences*, 118(2).

Hsiang, A. Y., Brombacher, A., Rillo, M. C., Mleneck-Vautravers, M. J., Conn, S., Lordsmith, S., Jentzen, A., Henehan, M. J., Metcalfe, B., Fenton, I. S., Wade, B. S., Fox, L., Meilland, J., Davis, C. V., Baranowski, U., Groeneveld, J., Edgar, K. M., Movellan, A., Aze, T., Dowsett, H. J., Miller, C. G., Rios, N., and Hull, P. M. (2019). Endless forams: 34,000 modern planktonic foraminiferal images for taxonomic training and automated species recognition using convolutional neural networks. *Paleoceanography and Paleoclimatology*, 34(7):1157–1177.

Jeong, K.-S., Joo, G.-J., Kim, H.-W., Ha, K., and Recknagel, F. (2001). Prediction and elucidation of phytoplankton dynamics in the Nakdong River (Korea) by means of a recurrent artificial neural network. *Ecological Modelling*, 146(1-3):115–129.

Jeong, K.-S., Kim, D.-K., Jung, J.-M., Kim, M.-C., and Joo, G.-J. (2008). Non-linear autoregressive modelling by temporal recurrent neural networks for the prediction of freshwater phytoplankton dynamics. *Ecological Modelling*, 211(3-4):292–300.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe. In *Proceedings of the ACM International Conference on Multimedia - MM 14*. ACM Press.

Jiang, Y., Balaban, M., Zhu, Q., and Mirarab, S. (2021). DEPP: Deep Learning Enables Extending Species Trees using Single Genes. *bioRxiv*.

Joutsijoki, H., Meissner, K., Gabbouj, M., Kiranyaz, S., Raitoharju, J., Ärje, J., Kärkkäinen, S., Tirronen, V., Turpeinen, T., and Juhola, M. (2014). Evaluating the performance of artificial neural networks for the classification of freshwater benthic macroinvertebrates. *Ecological Informatics*, 20:1–12.

Juneja, P. and Mitra, T. (2021). Auditing e-commerce platforms for algorithmically curated vaccine misinformation. In *Proceedings of the 2021 chi conference on human factors in computing systems*, pages 1–27.

Kahl, S., Wood, C. M., Eibl, M., and Klinck, H. (2021). BirdNET: A deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61:101236.

Kamilaris, A. and Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147:70–90.

Kern, A. D. and Schrider, D. R. (2018). diploS/HIC: An updated approach to classifying selective sweeps. *G3: Gemes, Genomes, Genetics*, 8(6):1959–1970.

Kessler, M. D., Yerges-Armstrong, L., Taub, M. A., Shetty, A. C., Maloney, K., Jeng, L. J. B., Ruczinski, I., Levin, A. M., Williams, L. K., Beaty, T. H., Mathias, R. A., Barnes, K. C., O'Connor, T. D., and Consortium on Asthma among African-ancestry Populations in the Americas (CAAPA) (2016). Challenges and disparities in the application of personalized genomic medicine to populations with African ancestry. *Nature Communications*, 7(1):12521.

Kim, B., Kim, H., Kim, K., Kim, S., and Kim, J. (2019). Learning Not to Learn: Training Deep Neural Networks With Biased Data. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9004–9012, Long Beach, CA, USA. IEEE.

Kiranyaz, S., Ince, T., Pulkkinen, J., Gabbouj, M., Ärje, J., Kärkkäinen, S., Tirronen, V., Juhola, M., Turpeinen, T., and Meissner, K. (2011). Classification and retrieval on macroinvertebrate image databases. *Computers in Biology and Medicine*, 41(7):463–472.

Kislov, D. E. and Korznikov, K. A. (2020). Automatic windthrow detection using very-high-resolution satellite imagery and deep learning. *Remote Sensing*, 12(7):1145.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Bussonnier, M., Frederic, J., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Abdalla, S., and Willing, C. (2016). Jupyter Notebooks—a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87–90.

Korznikov, K. A., Kislov, D. E., Altman, J., Doležal, J., Vozmishcheva, A. S., and Krestov, P. V. (2021). Using u-net-like deep convolutional neural networks for precise tree recognition in very high resolution RGB (red, green, blue) satellite images. *Forests*, 12(1):66.

Kroodsma, D. A., Mayorga, J., Hochberg, T., Miller, N. A., Boerder, K., Ferretti, F., Wilson, A., Bergman, B., White, T. D., Block, B. A., Woods, P., Sullivan, B., Costello, C., and Worm, B. (2018). Tracking the global footprint of fisheries. *Science*, 359(6378):904–908.

Körschens, M., Barz, B., and Denzler, J. (2018). Towards automatic identification of elephants in the wild.

Langmead, B. and Nellore, A. (2018). Cloud computing for genomic data analysis and collaboration. *Nature Reviews Genetics*, 19(4):208–219.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lenski, R. E., Rose, M. R., Simpson, S. C., and Tadler, S. C. (1991). Long-term experimental evolution in *Escherichia coli*. I. Adaptation and divergence during 2,000 generations. *The American Naturalist*, 138(6):1315–1341.

Li, J., Zhou, H., Wang, Z., and Jia, Q. (2020a). Multi-scale detection of stored-grain insects for intelligent monitoring. *Computers and Electronics in Agriculture*, 168:e105114.

Li, W., Buitenwerf, R., Munk, M., Bøcher, P. K., and Svenning, J.-C. (2020b). Deep-learning based high-resolution mapping shows woody vegetation densification in greater Maasai Mara ecosystem. *Remote Sensing of Environment*, 247:111953.

Li, W., Niu, Z., Shang, R., Qin, Y., Wang, L., and Chen, H. (2020c). High-resolution mapping of forest canopy height using machine learning by coupling ICESat-2 LiDAR with Sentinel-1, Sentinel-2 and Landsat-8 data. *International Journal of Applied Earth Observation and Geoinformation*, 92:102163.

Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., and Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, 21(6):335–346.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer.

Lins, E. A., Rodriguez, J. P. M., Scoloski, S. I., Pivato, J., Lima, M. B., Fernandes, J. M. C., da Silva Pereira, P. R. V., Lau, D., and Rieder, R. (2020). A method for counting and classifying aphids using computer vision. *Computers and Electronics in Agriculture*, 169:105200.

Liu, T., Chen, W., Wu, W., Sun, C., Guo, W., and Zhu, X. (2016a). Detection of aphids in wheat fields using a computer vision technique. *Biosystems Engineering*, 141:82–93.

Liu, X. and Song, H. (2020). Automatic identification of fossils and abiotic grains during carbonate microfacies analysis using deep convolutional neural networks. *Sedimentary Geology*, 410:105790.

Liu, Z., Gao, J., Yang, G., Zhang, H., and He, Y. (2016b). Localization and classification of paddy field pests using a saliency map and deep convolutional neural network. *Scientific Reports*, 6(1).

Liu, Z. Y.-C., Moxley, J. H., Kanive, P., Gleiss, A. C., Maughan, T., Bird, L., Jewell, O. J. D., Chapple, T. K., Gagne, T., White, C. F., and Jorgensen, S. J. (2019). Deep learning accurately predicts white shark locomotor activity from depth data. *Animal Biotelemetry*, 7(1):14.

Lozano, R., Gazave, E., dos Santos, J. P. R., Stetter, M. G., Valluru, R., Bandillo, N., Fernandes, S. B., Brown, P. J., Shakoor, N., Mockler, T. C., Cooper, E. A., Perkins, M. T., Buckler, E. S., Ross-Ibarra, J., and Gore, M. A. (2021). Comparative evolutionary genetics of deleterious load in sorghum and maize. *Nature Plants*, 7(1):17–24.

Mac Aodha, O., Cole, E., and Perona, P. (2019). Presence-only geographical priors for fine-grained image classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9596–9606.

Mac Aodha, O., Gibb, R., Barlow, K. E., Browning, E., Firman, M., Freeman, R., Harder, B., Kinsey, L., Mead, G. R., Newson, S. E., et al. (2018). Bat detective—deep learning tools for bat acoustic signal detection. *PLoS Computational Biology*, 14(3):e1005995.

Maekawa, T., Ohara, K., Zhang, Y., Fukutomi, M., Matsumoto, S., Matsumura, K., Shidara, H., Yamazaki, S. J., Fujisawa, R., Ide, K., Nagaya, N., Yamazaki, K., Koike, S., Miyatake, T., Kimura, K. D., Ogawa, H., Takahashi, S., and Yoda, K. (2020). Deep learning-assisted comparative analysis of animal trajectories with DeepHL. *Nature Communications*, 11(1):5316.

Malek, S., Salleh, A., Milow, P., Baba, M. S., and Sharifah, S. (2012). Applying artificial neural network theory to exploring diatom abundance at tropical Putrajaya Lake, Malaysia. *Journal of Freshwater Ecology*, 27(2):211–227.

Marques, A. C. R., Raimundo, M. M., Cavalheiro, E. M. B., Salles, L. F. P., Lyra, C., and Zuben, F. J. V. (2018). Ant genera identification using an ensemble of convolutional neural networks. *PLOS ONE*, 13(1):e0192011.

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., and Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 21(9):1281–1289.

Mathis, A., Schneider, S., Lauer, J., and Mathis, M. W. (2020). A primer on motion capture with deep learning: Principles, pitfalls, and perspectives. *Neuron*, 108(1):44–65.

McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction.

Miele, V., Dussert, G., Cucchi, T., and Renaud, S. (2020). Deep learning for species identification of modern and fossil rodent molars. *bioRxiv*.

Minin, E. D., Fink, C., Tenkanen, H., and Hiippala, T. (2018). Machine learning for tracking illegal wildlife trade on social media. *Nature Ecology & Evolution*, 2(3):406–407.

Mondal, M., Bertranpetit, J., and Lao, O. (2019). Approximate Bayesian computation with deep learning supports a third archaic introgression in asia and oceania. *Nature Communications*, 10(1):e246.

Montserrat, D. M., Bustamante, C., and Ioannidis, A. (2019). Class-conditional VAE-GAN for local-ancestry simulation.

Montserrat, D. M., Bustamante, C., and Ioannidis, A. (2020). Lai-Net: Local-ancestry inference with neural networks. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

Mundry, R. and Sommer, C. (2007). Discriminant function analysis with nonindependent data: consequences and an alternative. *Animal Behaviour*, 74(4):965–976.

Nath, T., Mathis, A., Chen, A. C., Patel, A., Bethge, M., and Mathis, M. W. (2019). Using DeepLabCut for 3D markerless pose estimation across species and behaviors. *Nature Protocols*, 14(7):2152–2176.

Nelson, M. J. and Hoover, A. K. (2020). Notes on Using Google Colaboratory in AI Education. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, pages 533–534, Trondheim Norway. ACM.

Norouzzadeh, M. S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M. S., Packer, C., and Clune, J. (2018). Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proceedings of the National Academy of Sciences*, 115(25):E5716–E5725.

Otto, S. P. and Day, T. (2011). *A biologist's guide to mathematical modeling in ecology and evolution*. Princeton University Press.

Ouellette, T. W., Shaw, J., and Awadalla, P. (2021). Using image-based haplotype alignments to map global adaptation of SARS-CoV-2. *bioRxiv*.

Park, Y. and Kellis, M. (2015). Deep learning for regulatory genomics. *Nature Biotechnology*, 33:825–826.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037.

Pereira, T. D., Aldarondo, D. E., Willmore, L., Kislin, M., Wang, S. S.-H., Murthy, M., and Shaevitz, J. W. (2019). Fast animal pose estimation using deep neural networks. *Nature Methods*, 16(1):117–125.

Pereira, T. D., Tabris, N., Li, J., Ravindranath, S., Papadoyannis, E. S., Wang, Z. Y., Turner, D. M., McKenzie-Smith, G., Kocher, S. D., Falkner, A. L., Shaevitz, J. W., and Murthy, M. (2020). SLEAP: Multi-animal pose tracking. *bioRxiv*.

Perez, M. F., Bonatelli, I. A. S., Romeiro-Brito, M., Franco, F. F., Taylor, N. P., Zappi, D. C., and Moraes, E. M. (2020). Coalescent-based species delimitation meets deep learning: Insights from a highly fragmented cactus system. *bioRxiv*, page e424219.

Poplin, R., Chang, P.-C., Alexander, D., Schwartz, S., Colthurst, T., Ku, A., Newburger, D., Dijamco, J., Nguyen, N., Afshar, P. T., Gross, S. S., Dorfman, L., McLean, C. Y., and DePristo, M. A. (2018). A universal SNP and small-indel variant caller using deep neural networks. *Nature Biotechnology*, 36(10):983–987.

Pratx, G. and Xing, L. (2011). GPU computing in medical physics: A review. *Medical Physics*, 38(5):2685–2697.

Priyadarshani, N., Marsland, S., and Castro, I. (2018). Automated birdsong recognition in complex acoustic environments: a review. *Journal of Avian Biology*, 49(5):e01447.

Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J., and Hughes, D. P. (2017). Deep learning for image-based cassava disease detection. *Frontiers in Plant Science*, 8:e1852.

Ramcharan, A., McCloskey, P., Baranowski, K., Mbilinyi, N., Mrisho, L., Ndalahwa, M., Legg, J., and Hughes, D. P. (2019). A mobile-based deep learning model for cassava disease diagnosis. *Frontiers in Plant Science*, 10:e272.

Rammer, W. and Seidl, R. (2019). Harnessing deep learning in ecology: An example predicting bark beetle outbreaks. *Frontiers in Plant Science*, 10:1327.

Rawat, W. and Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural Computation*, 29(9):2352–2449.

Reich, D., Price, A. L., and Patterson, N. (2008). Principal component analysis of genetic data. *Nature Genetics*, 40(5):491–492.

Roberts, M., Driggs, D., Thorpe, M., Gilbey, J., Yeung, M., Ursprung, S., Aviles-Rivero, A. I., Etmann, C., McCague, C., Beer, L., et al. (2021). Common pitfalls and recommendations for using machine learning to detect and prognosticate for covid-19 using chest radiographs and ct scans. *Nature Machine Intelligence*, 3(3):199–217.

Rumelhart, D. E., Durbin, R., Golden, R., and Chauvin, Y. (1995). Backpropagation: The basic theory. In Chauvin, Y. and Rumelhart, D. E., editors, *Backpropagation: Theory, Architectures, and Applications*, chapter 1, pages 1–34. Lawrence Erlbaum, Hillside, NJ.

Rustia, D. J. A., Chao, J.-J., Chung, J.-Y., and Lin, T.-T. (2019). An online unsupervised deep learning approach for an automated pest insect monitoring system. In *2019 Boston, Massachusetts July 7- July 10, 2019*. American Society of Agricultural and Biological Engineers.

Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., and Müller, K. R. (2021). Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278.

Sanchez, T., Cury, J., Charpiat, G., and Jay, F. (2020). Deep learning for population size history inference: design, comparison and combination with approximate Bayesian computation. *bioRxiv*.

Satoto, B. D., Utoyo, I., Rulaningtyas, R., and Khoendori, E. B. (2020). An improvement of gram-negative bacteria identification using convolutional neural network with fine tuning. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18(3):1397.

Schell, C. J., Dyson, K., Fuentes, T. L., Des Roches, S., Harris, N. C., Miller, D. S., Woelfle-Erskine, C. A., and Lambert, M. R. (2020). The ecological and evolutionary consequences of systemic racism in urban environments. *Science*, 369(6510):eaay4497.

Schmid, M. S., Aubry, C., Grigor, J., and Fortier, L. (2016). The LOKI underwater imaging system and an automatic identification model for the detection of zooplankton taxa in the Arctic Ocean. *Methods in Oceanography*, 15-16:129–160.

Schneider, S., Taylor, G. W., Linquist, S., and Kremer, S. C. (2019). Past, present and future approaches using computer vision for animal re-identification from camera trap data. *Methods in Ecology and Evolution*, 10(4):461–470.

Schofield, D., Nagrani, A., Zisserman, A., Hayashi, M., Matsuzawa, T., Biro, D., and Carvalho, S. (2019). Chimpanzee face recognition from videos in the wild using deep learning. *Science Advances*, 5(9).

Schrider, D. R. and Kern, A. D. (2018). Supervised machine learning for population genetics: a new paradigm. *Trends in Genetics*, 34(4):301–312.

Schuettpelz, E., Frandsen, P., Dikow, R., Brown, A., Orli, S., Peters, M., Metallo, A., Funk, V., and Dorr, L. (2017). Applications of deep convolutional neural networks to digitized natural history collections. *Biodiversity Data Journal*, 5:e21139.

Sejnowski, T. J. (2020). The unreasonable effectiveness of deep learning in artificial intelligence. *Proceedings of the National Academy of Sciences*, 117(48):30033–30038.

Sharif Razavian, A., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813.

Sheehan, S. and Song, Y. S. (2016). Deep learning for population genetic inference. *PLOS Computational Biology*, 12(3):e1004845.

Shen, Y., Zhou, H., Li, J., Jian, F., and Jayas, D. S. (2018). Detection of stored-grain insects using deep learning. *Computers and Electronics in Agriculture*, 145:319–325.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1):60.

Shu, L., Xu, H., and Liu, B. (2018). Unseen class discovery in open-world classification.

Sixt, L., Wild, B., and Landgraf, T. (2018). RenderGAN: Generating realistic labeled data. *Frontiers in Robotics and AI*, 5:66.

Smith, V. S. and Blagoderov, V. (2012). Bringing collections out of the dark. *ZooKeys*, 209:1–6.

Sorensen, D. and Gianola, D. (2002). *Likelihood, Bayesian, and MCMC Methods in Quantitative Genetics*. Springer New York.

Srinivasan, K., Duvvur, V., and Hess, D. (2018). Prediction of algal blooms in the Great Lakes through a convolution neural network of remote sensing data. *bioRxiv*, page e450551.

Stowell, D., Wood, M. D., Pamuła, H., Stylianou, Y., and Glotin, H. (2019). Automatic acoustic detection of birds through deep learning: the first bird audio detection challenge. *Methods in Ecology and Evolution*, 10(3):368–380.

Strydom, T., Catchen, M. D., Banville, F., Caron, D., Dansereau, G., Desjardins-Proulx, P., Forero, N., Higino, G., Mercier, B., Gonzalez, A., and et al. (2021). A primer on predicting species interaction networks (across space and time). *EcoEvoRxiv*.

Sulc, M., Picek, L., Matas, J., Jeppesen, T., and Heilmann-Clausen, J. (2020). Fungi recognition: A practical use case. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2316–2324.

Suvorov, A., Hochuli, J., and Schrider, D. R. (2019). Accurate inference of tree topologies from multiple sequence alignments using deep learning. *Systematic Biology*, 69(2):221–233.

Szubert, B., Cole, J. E., Monaco, C., and Drozdov, I. (2019). Structure-preserving visualisation of high dimensional single-cell datasets. *Scientific Reports*, 9.

Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Sweeney, S. J., Vercauteren, K. C., Snow, N. P., Halseth, J. M., Salvo, P. A. D., Lewis, J. S., White, M. D., Teton, B., Beasley, J. C., Schlichting, P. E., Boughton, R. K., Wight, B., Newkirk, E. S., Ivan, J. S., Odell, E. A., Brook, R. K., Lukacs, P. M., Moeller, A. K., Mandeville, E. G., Clune, J., and Miller, R. S. (2018). Machine learning to classify animal species in camera trap images: Applications in ecology. *Methods in Ecology and Evolution*, 10(4):585–590.

Tarca, A. L., Carey, V. J., Chen, X.-w., Romero, R., and Drăghici, S. (2007). Machine learning and its applications to biology. *PLoS Computational Biology*, 3(6):e116.

Teng, H., Cao, M. D., Hall, M. B., Duarte, T., Wang, S., and Coin, L. J. M. (2018). Chiron: translating nanopore raw signal directly into nucleotide sequence using deep learning. *GigaScience*, 7(5).

Terry, J. C. D., Roy, H. E., and August, T. A. (2020). Thinking like a naturalist: Enhancing computer vision of citizen science images by harnessing contextual data. *Methods in Ecology and Evolution*, 11(2):303–315.

The Theano Development Team, Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., et al. (2016). Theano: A Python framework for fast computation of mathematical expressions.

Thenmozhi, K. and Reddy, U. S. (2019). Crop pest classification based on deep convolutional neural network and transfer learning. *Computers and Electronics in Agriculture*, 164:104906.

Torada, L., Lorenzon, L., Beddis, A., Isildak, U., Pattini, L., Mathieson, S., and Fumagalli, M. (2019). ImaGene: a convolutional neural network to quantify natural selection from genomic data. *BMC Bioinformatics*, 20(S9).

Unger, J., Merhof, D., and Renner, S. (2016). Computer vision applied to herbarium specimens of German trees: testing the future utility of the millions of herbarium specimen images for automated identification. *BMC Evolutionary Biology*, 16(1).

Valan, M., Makonyi, K., Maki, A., Vondráček, D., and Ronquist, F. (2019). Automated taxonomic identification of insects with expert-level accuracy using effective feature transfer from convolutional networks. *Systematic Biology*, 68(6):876–895.

van Dam, E. A., Noldus, L. P., and van Gerven, M. A. (2020). Deep learning improves automated rodent behavior recognition within a specific experimental setup. *Journal of Neuroscience Methods*, 332:108536.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11):2579–2605.

Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., and Villéger, S. (2018). A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, 48:238–244.

Walter, M., Recknagel, F., Carpenter, C., and Bormans, M. (2001). Predicting eutrophication effects in the Burrinjuck Reservoir (Australia) by means of the deterministic model SALMO and the recurrent neural network model ANNA. *Ecological Modelling*, 146(1-3):97–113.

Wang, X., Cheng, J., and Wang, L. (2019a). Deep-reinforcement learning-based co-evolution in a predator–prey system. *Entropy*, 21(8):773.

Wang, Z., She, Q., and Ward, T. E. (2019b). Generative adversarial networks in computer vision: A survey and taxonomy. *arXiv*.

Wick, R. R., Judd, L. M., and Holt, K. E. (2019). Performance of neural network basecalling tools for Oxford Nanopore sequencing. *Genome Biology*, 20(1):129.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., Gonzalez-Beltran, A., Gray, A. J., Groth, P., Goble, C., Grethe, J. S., Heringa, J., 't Hoen, P. A., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S. J., Martone, M. E., Mons, A., Packer, A. L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S.-A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M. A., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., and Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3(1):160018.

Wittich, H. C., Seeland, M., Wäldchen, J., Rzanny, M., and Mäder, P. (2018). Recommending plant taxa for supporting on-site species identification. *BMC Bioinformatics*, 19(1).

Wolpert, D. H. and Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82.

Wu, X., Zhan, C., Lai, Y., Cheng, M.-M., and Yang, J. (2019). IP102: A large-scale benchmark dataset for insect pest recognition. In *IEEE CVPR*, pages 8787–8796.

Wäldchen, J. and Mäder, P. (2018). Machine learning for image based species identification. *Methods in Ecology and Evolution*, 9(11):2216–2225.

Ye, L. and Cai, Q. (2009). Forecasting daily chlorophyll a concentration during the spring phytoplankton bloom period in Xiangxi Bay of the Three-Gorges Reservoir by means of a recurrent artificial neural network. *Journal of Freshwater Ecology*, 24(4):609–617.

Yelmen, B., Decelle, A., Ongaro, L., Marnetto, D., Tallec, C., Montinaro, F., Furtlehner, C., Pagani, L., and Jay, F. (2019). Creating artificial human genomes using generative models. *bioRxiv*.

Younis, S., Weiland, C., Hoehndorf, R., Dressler, S., Hickler, T., Seeger, B., and Schmidt, M. (2018). Taxon and trait recognition from digitized herbarium specimens using deep convolutional neural networks. *Botany Letters*, 165(3-4):377–383.

Zhao, Z.-Q., Zheng, P., Xu, S.-t., and Wu, X. (2019). Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232.

Zhong, Y., Gao, J., Lei, Q., and Zhou, Y. (2018). A vision-based counting and recognition system for flying insects in intelligent agriculture. *Sensors*, 18(5):1489.

Zou, J., Huss, M., Abid, A., Mohammadi, P., Torkamani, A., and Telenti, A. (2018). A primer on deep learning in genomics. *Nature Genetics*, 51(1):12–18.

Zou, Z., Zhang, H., Guan, Y., and Zhang, J. (2020). Deep residual neural networks resolve quartet molecular phylogenies. *Molecular Biology and Evolution*, 37(5):1495–1507.